Generating Scientific Illustrations in Digital Books

Stefan Schlechtweg and **Thomas Strothotte**
Department of Simulation and Graphics
Otto-von-Guericke University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany

Abstract

In this paper we introduce what we call the "text illustration problem": Given a (scientific or technical) text which a user is studying and interacting with, as well as a collection of geometric models pertaining to the domain of the text. Produce customized, accompanying illustrations. We provide a framework for solving this problem and show how knowledge representation can enhance the quality of the illustrations generated. We demonstrate the feasibility of the approach through a prototypical system and show the results of applying it in the area of anatomy.

Introduction

The topic of digital books has recently stirred considerable interest. It is expected that new hardware will be developed which will allow the display of information on screens which behave much like paper. The hope is that this will lead to user interfaces which are ergonomically more pleasing than present-day computer screens. Users will be able to flip through electronic pages, much like the pages of a book.

The availability of such hardware in the future will require a redesign of the information which is actually to be displayed. On the one end of the spectrum, it will be possible to download pages as they otherwise appear in printed books. Indeed, while users today prefer to read and study long texts in printed books rather than on computers, the hope is that the new hardware will make reading computer output more attractive. On the other end of the spectrum, users will be able to interact with electronic books much like they interact with computers using screens today. New functionality will be needed to bridge this gap.

One area in which electronic books have the potential of providing added value over and above printed books is in customizing illustrations. Rather than a predefined illustration being embedded in the text at a fixed position, users should be able to ask for illustrations based on their current information needs. The illustrations should be generated taking into account the text at hand. This paper takes steps in this direction, showing how a new kind of image-text coupling based on a text model and geometric models pertaining to the domain of the text can be exploited.

The paper is organized as follows. First, background material on browsing through texts is given, and different kinds of image-text relations in printed books are discussed. Next, the topic of automatic text illustration is addressed and a solution is presented. Results of a prototypical system for interactive illustration of anatomical texts are given. Future work is outlined in the final section.

Background

Browsing through Texts, Short and Long

While addressing the issue of navigation in large information spaces, several visualization methods have emerged which are also suited for the purpose of browsing through long texts. The INFORMATION MURAL, as presented in (Jerding & Stasko 1998), is a two-dimensional reduced representation of an entire information space and fits completely within a display window or screen. It creates a miniature version of the information space using virtual attributes (color, etc.) to display its properties. It may be used as an overview of the structure of the observed information space and as a navigation tool.

To display long unstructured texts in a manner which resembles a stack of pages, the DOCUMENT LENS approach (Robertson & Mackinlay 1993) provides an innovative interface. Here, the pages are displayed as if the user looked at them through a magnifying glass. The currently selected page is shown in every detail, whereas adjacent pages are reduced in their size and perspectively warped. Also the WEBBOOK approach (Card, Robertson, & York 1996) provides an interesting interface for handling long texts.

The methods mentioned above concentrate on the inherent properties of the text and do not make use of any related information nor possibilities which emerge from including knowledge about the contents of the text into the interface. The method we propose for visualizing text is based on the interactive illustration of an otherwise long, unillustrated text. This raises the general issue of the interaction between texts and images, both in traditional print media and in multimedia systems. We shall therefore discuss these in turn.

Traditional Media: Atlases vs. Textbooks

Two fundamental forms of information presentation can be distinguished in traditional print media. On the one end of the spectrum, we have *atlases* whose primary focus is on graphics which illustrate the subject matter at hand in a very detailed way. The understanding of the subject is supported by labels and figure captions, and somewhat more rarely by additional texts and tables; thus text plays only a marginal role in such books. The attached labels have to be mentally integrated and—much more important—sorted out by the viewer to get the information he or she wants. Even more crucial for getting a complete impression of the subject matter is that one specific topic is dealt with in several images to convey the three-dimensionality from different viewpoints.

On the other end of the spectrum, we have *textbooks* which focus primarily on a verbal description of the subject matter at hand. For example, in medical textbooks, structure and functionality of the body parts and organs are described with an almost uniform terminology. The images used here accompany and illustrate the text and are thus less detailed than in atlases. The labeling of the text-book images is sparser than in atlases; in most cases only labels with a reference in the text are shown. Since images in textbooks concentrate on specific aspects, the number of images needed to illustrate a subject to a certain degree of completeness is rather high. This means that images may be spread over several pages, aggravating problems associated with the mental integration of all the information they contain. Furthermore, the reader needs to combine mentally both types of information, texts and graphics.

Both types of books, atlases and textbooks, have in common a high navigational effort to integrate textual and graphical information and to build up a mental model of the described contents. Flipping through pages as well as the use of different books at once is a standard—cumbersome and tiring—procedure for a student when learning anatomy.

Coupling Text and Graphics in Interactive Systems

Multimedia systems try to overcome some of the problems, especially those associated with the three-dimensional nature of the underlying geometric knowledge. They offer navigation facilities for 3D models by directly using them as a source for image generation and by offering the user possibilities to interact with the geometric model and thus change the viewpoint to whatever position he or she needs. Furthermore, text and graphics are directly linked to each other. Thus a user may request an image for a specific part of the text or an explanation of a specific part of the image by activating the link. However, practically all systems commercially available or reported on in the literature so far use predefined images or animation, and the user generally has no possibility to request a *customized* illustration which fits his or her intentions or reading history.

For interactive illustration systems, which often try to resemble the traditionally known media, we can also identify two major groups of systems. We will call systems which focus mainly on the graphics (and thus are similar to atlases) *graphics-driven*, and those which concentrate primarily on textual information (like textbooks) *text-driven*.

Graphics-Driven Systems Graphics-driven systems are widely used nowadays. Here a user interacts mainly with the graphical visualization to explore the information space. By interacting with the image, further data can be obtained, like object names, relationships, or additional facts. Texts are only used in small quantities and displayed as labels, figure captions or tables close to the image or in a separate window. As an example, this can be seen in "Sobotta. Atlas of Human Anatomy. CD-ROM edition". Here the very detailed images take almost the complete screen space. Picking an object in the image leads to the appearance or highlighting of a label or an entry in a supplementary table. Hence the image is in the center of interest and is used to obtain any information needed.

Graphics-driven systems can rely on a close connection between the textual information and the parts of the graphics they describe. Thus it is not astonishing that for this type of system a tight coupling between graphics and text is established where the graphical model plays the central role. The ZOOMILLUSTRATOR by PREIM et al. is an excellent example for this. Here, short texts are used to label an image which in turn is generated from a 3D model (Preim, Raab, & Strothotte 1997). The user selects the parts he or she wants to be labeled and the ZOOMILLUSTRATOR generates and displays the labels. Interaction with the labels yields a more or less detailed description. The size and placement of the labels is hereby controlled by a fisheye zoom algorithm (Furnas 1986). An advantage is that because of this rigid link, search operations in the information space are kept to a minimum. However, the biggest drawback is that this strategy can not be used for long texts where more than one paragraph applies to one part in the image. Also, to change the texts here means to change the model as a whole.

Not only textual labels but all kinds of information can be attached at a graphical model in the system VOXELMAN. Here, voxel models serve as a basis for the graphics generation and to each voxel pointers into different information spaces are attached (Höhne *et al.* 1994). All information spaces are linked to each other and form a kind of a semantic network so that a wide variety of information can be explored and displayed on the user's demand. This approach is promising since it enables to combine many different types of data, though it is cumbersome to use for long texts. Furthermore, the information space is built on (artificial) physical parts of the model (the voxels) instead of logical parts (organs, body parts) and is thus very expensive to create.

Text-Driven Systems Text-driven systems, by contrast, are still rather underestimated as to their usefulness in learning environments. An example for such a system is the "Microsoft Encarta Encyclopedia". Here the main information source are textual descriptions of things, events, persons, etc. Those texts are connected with predefined images, sounds, or video clips. As the user scrolls through the text, this related information is displayed depending on the topic currently shown in the text. If the topic changes (for instance, if the next entry starts), the image (or whatever is currently displayed) also changes. Hence all information presented to the user is kept consistent.

A major problem when relying on textual information in on-line systems is the amount of text being displayed. Due to screen space limitations only a very small part can be shown at any one point in time. As a metaphor for flipping pages, scrolling through the text by using scrollbars and other interaction facilities is not very well suited. Reading long texts on a computer screen differs from reading text on paper in many respects. Experiments like the one done by O'HARA and SELLEN show that the standard interaction facilities offered so far do not help to concentrate on the task at hand but distract the user by drawing his or her attention to the interaction which has to be performed (O'Hara & Sellen 1997).

When building a text-driven illustration system, we consider the following points as being important:

- *Provide integration of text and images:* Textual and graphical information are to be incorporated in a way to make the relationship between them immediately clear to the user.

- *Facilitate interaction:* Interaction facilities with the illustration should at least support change of viewpoint and zooming.

- *Provide multiple access to information:* Access to information should not be restricted to one particular interaction method; instead, many possible ways should be offered to get a specific piece of information.

- *Provide flexible levels of detail:* The amount of information (especially textual information) provided to the user should be controllable by the user, though in certain situations the system itself may decide that more or less detailed information will be presented.

Our goal is to design a system which enables users to browse through long texts to get an overview and to find quickly sections of interest and interrelations between them. In particular, we emphasize the coupling of text and custom generated images, as well as providing access to the text also through the images.

Illustrating Texts

For solving what we have called the *text illustration problem*, i.e., the graphical communication of the contents of a given text, we follow the approach of a text-driven system as outlined above.

On the textual side, we provide an interface to navigate through the text as it is already familiar to the users for instance from web browsing software. Additional interactive aids like adding annotations and changing the level of detail at which the text is displayed shall help to reduce the effort to navigate within the textual domain. The graphical illustrations are created from a given 3D model (in our case a surface model, however, the techniques introduced here can be extended to any kind of geometry representation) and rendered in an interactive viewer. Here, standard interaction techniques like rotation, translation, and zoom are available to explore the model.

A new quality is achieved by combining both models and thus creating an architecture where navigation is possible in *both* models and also affects the display of *both* models. The basis for such a system is an architecture as shown in Figure 1.

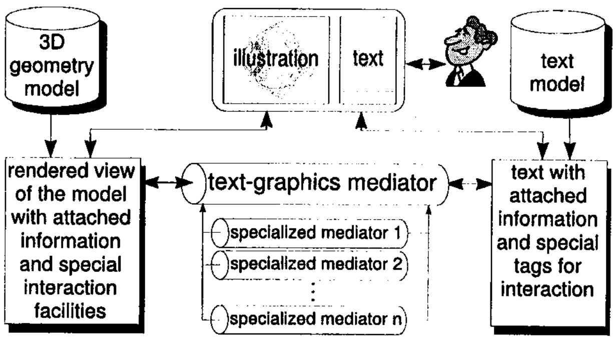

Figure 1: System architecture

The Text Model

To realize the additional navigation facilities, the text model has to contain more than the pure ASCII representation of the text. The inherent structure given by chapters, sections, and paragraphs has to be represented as well as formatting instructions (formatting is often used to emphasize/deemphasize certain passages) and possibly cross-references. To incorporate all these information in the text while still being able to process the text algorithmically, markup languages are the tool to be chosen. HTML is widely recognized as the language for the World Wide Web but rather inflexible for other tasks. Extensible markup languages like SGML or XML serve this purpose well. In addition to the above mentioned features as there are document structuring and emphasis, using XML makes it possible to also mark such entities in the text which represent graphical objects and thus to establish links between the text and the 3D model.

The markup of words in the text which are used as such links can be done automatically for certain domains. Medicine (anatomy) is a very good example for such a domain. In such texts, a standardized terminology (Paris Nomina Anatomica) is used which is agreed upon by an independent committee and which almost all authors use. (Actually, there exist several nomenclatures which are well documented and which are rarely used together). Another example is architecture or—to a lesser extent—engineering. If there is no possibility to do an automatic markup, the user should be provided with special editors to ease that task.

The Graphical Model

There are several requirements for the 3D geometry model if they are intended to be used in a setup described here. We use polygonal surface models, although the method is not limited to those. Any other model will suffice as long as it consists of clearly recognizable and distinguishable objects. This is also the strongest requirement for the models we have. Each object (typically a collection of polygonal faces, surface patches or voxels) has to be identifiable by a unique ID which is in the case of anatomy derived from the object's name in the standardized nomenclature. This ID will serve as the connection to the text model and will be used within the interaction process to identify objects and handle them correctly.

Relations which can be expressed in the geometric model can be used for structuring the information space in a similar manner as within the text model. Here especially *part-of* relations are often included in a geometry model. The object groups which are constructed in this way have to be assigned an ID as well. However, the user should not rely on the presence of such object hierarchies since different model sources make it difficult to establish a standard of what has to be in the model.

Connecting Both Worlds

The connection between both models is of crucial importance since it determines the interactive possibilities which are available for exploring the combined information space. Hence a monolithic model containing both parts would only be a hindrance for the development of new interaction techniques. A loose coupling between graphical and textual model as illustrated in Figure 1 is thus the basis for the concept presented here.

Connecting the text to the graphical model relies on the identifiers in the geometry model. Special entities, so called ***Text-Graphics-Mediators*** serve as "brokers" which translate interaction events on one side into changes on the other side. For each combination of interaction event and system reaction, a special mediator has to be available. A few examples should make clear how this concept works.

A user selects a word in the text by clicking on it using a pointing device (like a mouse). The mediator which is responsible for this kind of interaction checks whether this word has an associated object id in the geometry (encoded by special markups in the text). If so, the respective object is identified and marked as highlighted. The geometry viewer then displays this object appropriately. A different mediator, also responsible for selecting single words in the text, may change the viewing matrices in the geometry viewer in a way that the selected object becomes visible and recognizable. As a second example, selecting an object directly in the graphical view may result in search operations on the text so that a paragraph of text which is connected to this object is displayed. Also multiple selections can be supported—as for instance by color coding all those objects which appear in the currently visible text portion. The latter approach leads to a reduction in the search effort necessary when browsing through a text, looking for a specific topic. Since the graphics change in accordance to the objects mentioned in the visible text portion, a user no longer actually needs to read the text; instead observations of the generated graphical animation help to find at least a text part which has something to do with the object in question.

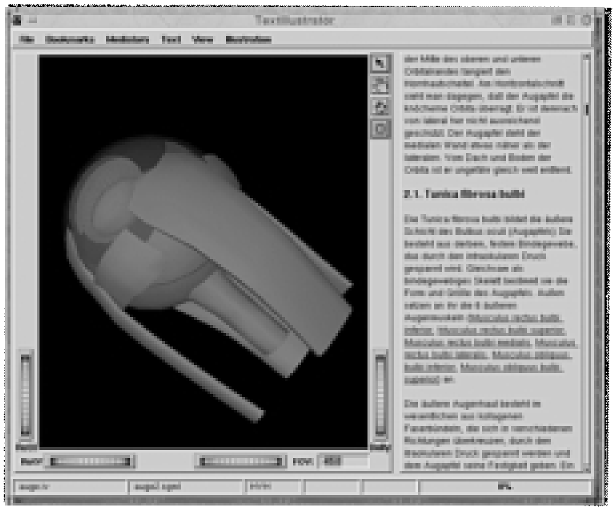

Figure 2: Screen shot of an interactive illustration system using the concepts described in this paper.

Using Interaction to Derive Visualization Parameters

Besides the navigation resulting in changes to the viewpoint which have been described so far, user interaction can also be used to determine parameters for visualizations being created from the geometric model. Thus it will become possible to create images which reflect not only a geometry but also the contents of a given text and moreover the interest of the user in selected parts of this geometry (or text).

When interacting with this combined information space, the user shows his or her interest in certain objects. The degree of interest differs from very high—when the used directly interacts with only *one* object—to lower when more objects are included in the interaction process. To create illustrations which reflect those different degrees of interest with different visualization styles, a quantization of this degree of interest has to be found. FURNAS developed such a concept in connection with fisheye zoom algorithms (cf. (Furnas 1986)) which can be adapted for the situation at hand.

Each object has a "pre-defined" importance value which results from its position in the information space—the place in the hierarchy given in the geometry model or any importance information which can be derived from an analysis of the text, for instance. This *à priori importance* API is thus the basis for any degree of interest calculation and the degree of interest DOI for an object o is initially set to this API:

$$DOI_0(o) = API(o)$$

When users interact with an object, the degree of interest of this particular object increases. But also other objects experience a change in DOI since all objects which are no longer in the center of interest (which are no longer interacted with) may decrease in their own degree of interest. To formalize this, the DOI after each interaction step is calculated as follows:

$$DOI_n(o) = f(DOI_{n-1}(o), d(o, FP))$$

The function d describes the distance of an object o from the focus point FP of the interaction, i.e., all those objects with which the user currently interacts directly. This distance is not only meant topographically but also conceptually in order to realize, for instance, changes over time. Finally, the function f combines the old DOI value with this distance function. f depends on the kind of interaction performed so that different interactions lead to different results. This is necessary because a *direct* interaction with a single object—for instance, by selecting an object with the mouse – shows more interest in this particular object as for instance the selection of a group of objects.

Using this theoretical framework, a DOI value can be calculated for each object in the scene. This DOI value represents the user's interaction as well as an overall importance of each object in the information space (given by the API values). To show these values graphically, a mapping onto visualization parameters has to be found. So-called *presentation variables* are introduced where a change in a value of this variable results in a change of the visualization. This concept was mentioned the first time in (Noik 1994). For our purposes, the most important presentation variable is the *visualization style* which determines if an object is rendered photorealistically, more schematically, as a line drawing or if it is visualized as an rather abstract diagram. Based on this visualization style, a rough classification of objects according to their importance is achieved.

Within each of the classes representing different visualization styles, other presentation variables determine the exact rendering of an object by, for instance, choosing the color or the line style and line thickness, or some hatching parameters. Figure 3 shows an example stemming from such a visualization. Here also, the contents of the text is taken into account. The text portion to which this image is generated deals with the six muscles which move the eyeball. Hence all other objects are shown with much less detail as simple contour drawings. The emphasis of the optical nerve results from the interaction history.

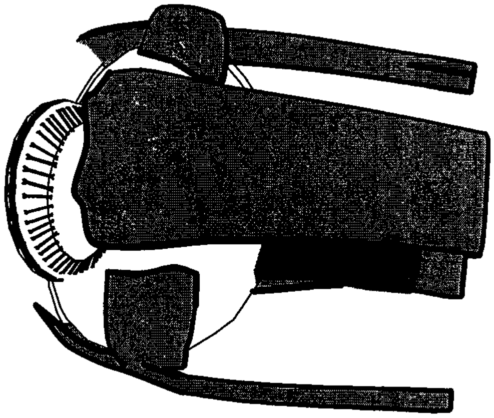

Figure 3: Using different visualization styles to show different user interest.

Adding Knowledge to the Illustration

So far in this paper we have shown how to solve the so-called *text illustration problem* as stated above. We assume that a structured geometric model is available for the objects discussed in the text. The relations between terms in the text and parts of the geometric model are established via unique IDs. Being able to solve this text illustration problem has some implications. First, we can create customized illustrations which do not only reflect the contents of a given text and/or the intentions of the scientist-author but also the user's interest in certain parts of the information space. Second, the main problem using this approach is that the illustrations only reflect *relations between objects* if they are explicitly modeled either in the geometry or in the markup of the text. Thus all information over and above that already containing in the models has to be brought in manually.

If we consider, for instance, medical or anatomical texts, many additional relations are given in these textual descriptions. Especially anatomical texts use a highly standardized terminology in such cases. (As an example the description of a muscle has in many cases the form: "Muscle X has its origin in ... and inserts into ...). Parsing the text and including such relations into the information space brings us in a position to use additional knowledge in the creation of the illustration. In connection with the interaction analysis and the degree of interest approach this enables the creation of highly specialized illustrations suited for use in textbooks or other learning aids.

Modeling Additional Knowledge

To successfully use this additional knowledge on relations between objects, we have to incorporate these relations in the information space. This was done in a first step by manually creating a knowledge base from a given anatomical text. For this, a 70 page text from an anatomy textbook (Rogers 1992) was prepared by K. Hartmann. Included in this process was

- finding all "medical objects" which may have an equivalent in the geometry,
- classifying these objects according to classification schemes used in the domain,
- finding all relations between the objects, and
- classifying these relations according to their type.

As a result, the knowledge is modeled in a semantic net with nodes representing the objects and edges representing relations. Due to the different types of relations, different weights are assigned to the edges in order to control the "tightness" of the connection described by a certain relation. The final knowledge base contains about 500 concepts and also about 500 relations. This number takes into account only text that describes the anatomy of the human leg below (but not including) the knee joint.

Interaction

When interacting with a specific object, these modeled relations should lead to a change in the degree of interest of related objects. Here we make extensive use of the weight values assigned to the semantic network's edges. There are relations which connect two objects very closely (the respective edge has a high weight value) while others are only a very weak conceptual link (low weight value).

When interacting with the information space, for each object the new degree of interest is computed as described above. However, this computed DOI is not the final DOI assigned to the object. In addition to the changes invoked directly by the interaction, the DOI also changes due to the relations with other objects. Each new DOI value is distributed in the network along the edges according to the weights assigned to each edge. The propagation algorithm increases for instance the DOI of bones which are the origin or insertion of a muscle with which the interaction is performed. Using this method, the relations modeled in the knowledge base also drive the visualization and hence are visualized in the illustration.

For the sake of completeness, we shall now describe the propagation algorithm; for more detail see (Hartmann *et al.* 1999). Interaction with (one or more) objects in the illustration or the text leads to a change of DOI of this object. This change *activates* a node in the semantic net by a so-called *dominance value* which represents the new DOI. Each node then consumes a certain amount of this dominance value and propagates the rest along the edges starting at the activated node. How much of the dominance is propagated along an edge is determined by the relation between the weight values of all edges. This process is recursively repeated until either all dominance is consumed by the nodes or until no further edges leading to other nodes exist. Finally, the dominance values received by each node are translated in new DOI values for the respective objects.

Results

The following Figure 4 shows a first result of the inclusion of additional knowledge into the illustration. The user has selected one muscle (Musculus peroneus longus) directly in the illustration thus showing a very high interest in this object. Due to the "has origin" and "has insertion" relations in the knowledge base, several bones are assigned a certain degree of interest. They are then visualized in a style which emphasizes them over all objects which are unrelated to the originally selected muscle.

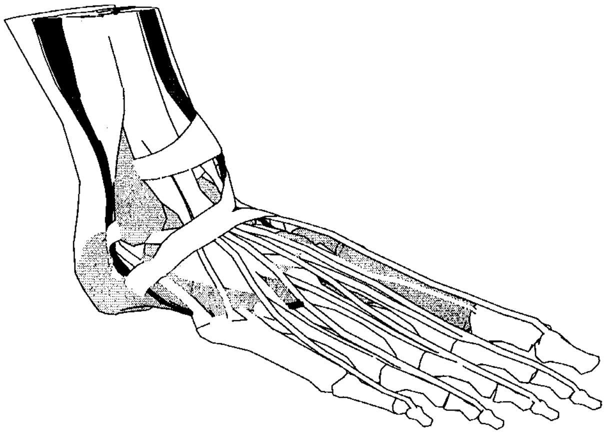

Figure 4: Using modeled relations between the domain objects reveals correlations to the viewer. The highest DOI is visualized using dark gray and a strong line showing the muscle. Directly related objects (having a smaller DOI) are drawn in dark or light gray. For completely unrelated objects, only the contour is visualized.

Due to the knowledge representation added to the pure text illustration it is possible to make use of a broader range of graphical techniques, as is evident from Figure 4. The additional knowledge processing gives the opportunity to distribute the interest over conceptually related objects thus exploiting to a larger extent the capabilities of non-photorealistic rendering.

We wish to emphasize that up to now the knowledge base has been constructed manually. It is an interesting open problem to try to use and develop techniques for automatically constructing a knowledge base to the specific extent necessary for high quality text illustration. Up to now it is unclear how difficult this process actually is, particularly compared to deriving a general knowledge base from a given text or, even more general, for the domain at hand.

Conclusions

Visualizing the contents of a text requires not only identification of model parts and highlighting them graphically. Important is also the visualization of *relations* which are rarely specified in the geometric model. Here, more information has to be extracted from the given data—especially the text. Although everything should work automatically, the user's interaction is of particular importance to drive the process of selecting the information in which the user is most interested.

In this paper we have presented a solution to what we have called the *text illustration problem.* Illustrations are generated based on the contents of a given text and also reflect the user's interest in certain objects. The basis for this is a geometric model with uniquely identifiable objects and a text which has been marked up with respect to the generation of an illustration from this geometry. We have seen that in many application areas—especially medicine, architecture, and engineering—the text markup can be done at least semi-automatically.

By solving the text illustration problem we have devised some principles and techniques which form the basis for the automatic generation of more sophisticated illustrations. These include also the visualization of *relations* between the observed objects. A knowledge base in which these relations are modeled is used to distribute the calculated degree-of-interest values accordingly. In future, we would like the contents of the knowledge base to be extracted automatically from the text (or from more texts from the given application area).

There is still the question of the tradeoff between the completeness of the knowledge base and the quality of the illustration. It is clear so far that including modeled knowledge yields an added value in the illustration. To what extent an increase in the amount of available knowledge also increases the quality of the illustration is not yet clear.

The set of visualization techniques as used in this paper and presented in (Schlechtweg *et al.* 1998) gives a good foundation for automatic illustration generation. We have concentrated so far on visualization techniques which do not include animation or automatic changes in camera position and parameters. Being able to choose automatically a camera position which best shows the objects in question or even to generate an animation which "illustrates" a process or spatial data will make use of the possibilities offered in multimedia systems today.

User interaction, especially with the text, is also crucial. Here new ways of presenting the text to the user as well as interacting with the text (using all its inherent properties) are to be developed. Furthermore, the problem of annotating electronic documents has become more difficult in the area of knowledge supported illustration. If already the question of how to add an annotation to an electronic text is not easy to solve, the question of how to integrate the user's annotations to the knowledge base is even further away from being answered.

This paper has shown a few steps toward integrating 3D models, text, and additional knowledge and thus toward a new quality which exposes electronic books from being pure copies of traditional media.

Acknowledgments

The authors wish to thank all who contributed to this paper by discussing the topic and sharing ideas. A very special thank to Knut HARTMANN who developed and implemented the algorithms needed to make our graphics "smarter".

References

Card, S. K.; Robertson, G. G.; and York, W. 1996. The WebBook and the Web Forager: An Information Workspace for the World-Wide-Web. In *Proceedings of CHI'96 Conference on Human Factors in Computing Systems (New Orleans, April 1996)*, 111–117. New York: ACM SIGCHI.

Furnas, G. W. 1986. Generalized Fisheye Views. In *Proceedings of CHI'86 Conference on Human Factors in Computing Systems (Boston, April 1986)*, 16–23. New York: ACM SIGCHI.

Hartmann, K.; Krüger, A.; Schlechtweg, S.; and Helbing, R. 1999. Interaction and Focus: Towards a Coherent Degree of Detail in Graphics, Captions, and Text. In Deussen, O.; Hinz, V.; and Lorenz, P., eds., *Proceedings of Simulation und Visualisierung '99*, 127–137. Erlangen · Ghent: SCS Europe.

Höhne, K.-H.; Pommert, A.; Riemer, M.; Schiemann, T.; Schubert, R.; and Tiede, U. 1994. Medical Volume Visualization Based on 'Intelligent Volumes'. In *Scientific Visualization. Advances and Challanges.* London: Academic Press. 21–35.

Jerding, D. F., and Stasko, J. T. 1998. The Information Mural: A Technique for Displaying and Navigating Large Information Spaces. *IEEE Transactions on Visualization and Computer Graphics* 4(3):257–271.

Noik, E. 1994. A Space of Presentation Emphasis Techniques for Visualizing Graphs. In Davis, W. A., and Joe, B., eds., *Proceedings of Graphics Interface'94 (Banff, May 1994)*, 225–234. Toronto: Canadian Computer-Human Communications Society.

O'Hara, K., and Sellen, A. 1997. A Comparison of Reading Paper and On-Line Documents. In *Proceedings of CHI'97 Conference on Human Factors in Computing Systems (Atlanta, March 1997)*, 335–342. New York: ACM SIGCHI.

Preim, B.; Raab, A.; and Strothotte, T. 1997. Coherent Zooming of Illustrations with 3D-Graphics and Text. In *Proceedings of Graphics Interface'97 (Kelowna, Kanada, May 1997)*, 105–113. Toronto: Canadian Computer-Human Communications Society.

Robertson, G. G., and Mackinlay, J. D. 1993. The Document Lens. In *UIST: Proceedings of the ACM Symposium on User Interface Software and Technology '93*, 101–108. New York: ACM.

Rogers, A. W. 1992. *Textbook of Anatomy*. Edinburgh: Churchill Livingstone.

Schlechtweg, S.; Schönwälder, B.; Schumann, L.; and Strothotte, T. 1998. Surfaces to Lines: Rendering Rich Line Drawings. In Skala, V., ed., *Proceedings of WSCG'98, The 6th International Conference in Central Europe on Computer Graphics and Visualization (Pilzeň, February 1998)*, 354–361.